Policy invariance under reward transformations: Theory and application to reward shaping

Andrew Y. Ng, Daishi Harada, Stuart Russell
Computer Science Division
University of California, Berkeley
Berkeley CA 94720
{ang,daishi,russell}@cs.berkeley.edu

Abstract

This paper investigates conditions under which modifications to the reward function of a Markov decision process preserve the optimal policy. It is shown that, besides the positive linear transformation familiar from utility theory, one can add a reward for transitions between states that is expressible as the difference in value of an *arbitrary* potential function applied to those states. Furthermore, this is shown to be a necessary condition for invariance, in the sense that any other transformation may yield suboptimal policies unless further assumptions are made about the underlying MDP. These results shed light on the practice of *reward shaping*, a method used in reinforcement learning whereby additional training rewards are used to guide the learning agent. In particular, some well-known "bugs" in reward shaping procedures are shown to arise from non-potential-based rewards, and methods are given for constructing shaping potentials corresponding to distance-based and subgoal-based heuristics. We show that such potentials can lead to substantial reductions in learning time.

1 Introduction

In sequential decision problems, such as are studied in the dynamic programming and reinforcement learning literatures, the "task" is represented by the *reward function*. Given the reward function and a model of the domain, the optimal policy is determined. An elementary theoretical question that arises is this: What freedom do we have in specifying the reward function, such that the optimal policy remains unchanged?

In the field of utility theory, which studies primarily single-step decisions, the corresponding question for the utility function can be answered very simply. For single-step decisions without uncertainty, any monotonic transformation on utilities leaves the optimal decision unchanged; with uncertainty, only positive linear transformations are allowed [von Neumann and Morgenstern, 1944]. These results have important implications for designing evaluation functions in games, eliciting utility functions from humans, and many other areas.

To our knowledge, the question of policy invariance under reward function transformations has not been fully explored for sequential decision problems.[1] Policy-preserving transformations are important at least in these areas:

- The task of *structural estimation* of MDPs [Rust, 1994] involves recovering the model and reward function from observed optimal behavior. (See also the discussion of *inverse reinforcement learning* in [Russell, 1998].) Policy-preserving transformations determine the extent to which a reward function can be recovered.

- The practice of *reward shaping* in reinforcement learning consists of supplying additional rewards to a learning agent to guide its learning process, beyond those supplied by the underlying MDP. It is important to understand the impact of shaping on the learned policy.

[1] Some results are known for approximate invariance: if rewards are perturbed by at most ε, the new policy's value is within $2\varepsilon/(1-\gamma)$ of the original optimal policy [Singh and Yee, 1994, Williams and Baird, 1994].

This paper focuses primarily on reward shaping, which has the potential to be a very powerful technique for scaling up reinforcement learning methods to handle complex problems [Dorigo and Colombetti, 1994, Mataric, 1994, Randløv and Alstrøm, 1998]. (Similar ideas have arisen in the animal training literature; see [Saksida et al., 1997] for a discussion.) Often, a very simple pattern of extra rewards suffices to render straightforward an otherwise completely intractable problem.

To see why policy invariance is important in shaping, consider the following examples of bugs that can arise: [Randløv and Alstrøm, 1998] describes a system that learns to ride a simulated bicycle to a particular location. To speed up learning, they provided positive rewards whenever the agent made progress towards the goal. The agent learned to ride in tiny circles near the start state because no penalty was incurred for riding away from the goal. A similar problem occurred with a soccer-playing robot being trained by David Andre and Astro Teller (personal communication). Because possession in soccer is important, they provided a reward for touching the ball. The agent learned a policy whereby it remained next to the ball and "vibrated," touching the ball as frequently as possible. These policies are clearly not optimal for the original MDP.

These examples suggest that the shaping rewards must obey certain conditions if they are not to mislead the agent into learning suboptimal policies. The difficulty with positive-reward cycles leads one to consider rewards derived from a conservative potential—that is, the reward for executing a transition between two states is (essentially) the difference in the value of a potential function applied to each state. It turns out that not only is this a sufficient condition for guaranteeing policy invariance under reward transformations, but that, assuming no prior knowledge of the MDP, this is also a *necessary* condition for being able to make such a guarantee. Section 2 gives the definitions needed to state this claim precisely, and Section 3 states and proves the claim. Section 4 shows how to construct shaping potentials of various kinds and demonstrates their efficacy in speeding up learning on some simple domains. Finally, Section 5 connects our results to existing algorithms such as Advantage learning [Baird, 1994] and λ-policy iteration [Bertsekas and Tsitsiklis, 1996], and closes with discussion and future work.

2 Preliminaries

2.1 Definitions

In this section, we provide some of the definitions used throughout the paper, focusing on the case of finite-state Markov decision processes (MDPs). Shaping is of interest to us in both finite-state and infinite-state problems, but the underlying MDP theory for the infinite-state case is significantly more difficult, even in the absence of shaping. Nevertheless, our analysis and methodology are easily generalized from the finite to the infinite-state space case once the underlying MDP theory is laid out, and we will mention this again later; but for now, let us start our definitions with explicitly considering only finite-state domains.

A (finite-state) **Markov decision process (MDP)**, is a tuple $M = (S, A, T, \gamma, R)$, where: S is a finite set of **states**; $A = \{a_1, \ldots, a_k\}$ is a set of $k \geq 2$ **actions**; $T = \{P_{sa}(\cdot) | s \in S, a \in A\}$ are the next-state **transition probabilities**, with $P_{sa}(s')$ giving the probability of transitioning to state s' upon taking action a in state s; $\gamma \in (0, 1]$ is the **discount factor**; and R specifies the reward distributions. For simplicity, we will assume rewards are deterministic, in which case R is a bounded real function called the **reward function**. In the literature, reward functions are typically written $R : S \times A \mapsto \mathbb{R}$, with $R(s, a)$ being the reward received upon taking action a in state s. Though we will often write reward functions in this form, we will also allow a more general form, $R : S \times A \times S \mapsto \mathbb{R}$, with $R(s, a, s')$ being the reward received upon taking action a in state s and transitioning to state s'.

Given a fixed set of actions A, a **policy** over a set of states S is any function $\pi : S \mapsto A$. Note that policies are defined over states and not over MDPs, so the same policy may be applied to two different MDPs so long as the two MDPs use the same states and actions. Given any policy π over states S and any MDP $M = (S, A, T, \gamma, R)$ using the same states and actions, we may then define the **value function** V_M^π, which evaluated at any state s gives $V_M^\pi(s) = \mathrm{E}[r_1 + \gamma r_2 + \gamma^2 r_3 + \ldots ; \pi, s]$, where r_i is the reward received on the ith step of executing the policy π from state s, and the expectation is over the state-transitions taken upon executing π. We then define the **optimal value function** to be $V_M^*(s) = \sup_\pi V_M^\pi(s)$, the Q**-function**, evaluated at any $s \in S, a \in A$ as

$$Q_M^\pi(s, a) = \mathrm{E}_{s' \sim P_{sa}(\cdot)} \left[R(s, a, s') + \gamma V_M^\pi(s') \right] \quad (1)$$

(where the notation $s' \sim P_{sa}(\cdot)$ means that s' is drawn according to the distribution $P_{sa}(\cdot)$), and the **optimal**

Q**-function** as $Q^*_M(s,a) = \sup_\pi Q^\pi_M(s,a)$. Finally, we define the **optimal policy** for an MDP M as $\pi^*_M(s) = \arg\max_{a\in A} Q^*_M(s,a)$. The optimal policy may not be unique, and we more generally say a policy π is optimal in M if $\pi(s) \in \arg\max_{a\in A} Q^*_M(s,a)$ for all $s \in S$. Lastly, when the context MDP is clear, we may also drop the M-subscript, and write V^π rather than V^π_M, etc.

We also need some (largely standard) regularity conditions so as to make sure all of the above definitions make sense. For undiscounted ($\gamma = 1$) MDPs, we assume that S contains a distinguished state s_0 called an **absorbing state**, so that the MDP "stops" after a transition into s_0, with no further rewards thereafter. Moreover, again for undiscounted MDPs, we assume all policies are **proper**, meaning that upon executing any policy starting from any state, we will with probability 1 eventually transition into s_0. Since this is really a condition on T, we will in this paper say the *transition probabilities* T are proper if this condition holds. Discounted MDPs have no corresponding absorbing state and are always infinite-horizon; note therefore that for them, we can write $S - \{s_0\} = S$.

The above were the standard regularity conditions needed for MDPs with *finite* state spaces (see, e.g. [Sutton and Barto, 1998]), which is the case which we had explicitly said we would focus on. For MDPs with infinite state spaces, more would be needed: for example, in the undiscounted case, the expectation in our definition of $V^\pi_M(s) = \mathrm{E}[r_1 + \gamma r_2 + \gamma^2 r_3 + \dots; \pi, s]$ may not even exist.[2] These issues need to be properly addressed before we can even define things such as optimal policies, and excellent sources for this material include [Bertsekas, 1995, Hernández-Lerma, 1989, Bertsekas and Shreve, 1978]. But unfortunately, explaining the infinite $|S|$ case in full generality would require more measure theory than we wish to delve into here, and we only comment that, with the appropriate generalizations of the required regularity conditions on the MDP, all of our results are easily generalized to the infinite $|S|$ case. Throughout this paper, we will however continually draw links to how the results may be proved for infinite $|S|$, though we defer the more general proofs for infinite $|S|$ to the full paper. For now, we note only that for the infinite-state case, an important and useful condition is that the reinforcements are bounded in absolute value; this will be mentioned again later in the paper.

[2]This is in a similar sense to the "mean" of a Cauchy distribution not existing.

2.2 Shaping Rewards

In this section, we introduce our formal framework of shaping rewards. Intuitively, we are trying to learn a policy for some MDP $M = (S, A, T, \gamma, R)$, and we wish to help our learning algorithm by giving it additional "shaping" rewards which will hopefully guide it towards learning a good (or optimal) policy faster. To formalize this, we assume that, rather than running our reinforcement learning algorithm on $M = (S, A, T, \gamma, R)$, we will run it on some *transformed* MDP $M' = (S, A, T, \gamma, R')$, where $R' = R + F$ is the reward function in the transformed MDP, and $F : S \times A \times S \mapsto \mathbb{R}$ is a bounded real-valued function called the **shaping reward function**. (Similar to R, the domain of F for the undiscounted case should strictly be $S - \{s_0\} \times A \times S$, but we will not be overly pedantic about this point for now.) So, if in the original MDP M we would have received reward $R(s,a,s')$ for transitioning from s to s' on action a, then in the new MDP M' we would receive reward $R(s,a,s') + F(s,a,s')$ on the same event.

For any fixed MDP and assuming additive, memoryless shaping reward functions, this $R' = R + F$ is the most general possible form of shaping rewards.[3] Moreover, they cover a fairly large range of possible shaping rewards one might come up with. For example, to encourage moving towards a goal, a shaping-reward function that one might choose is $F(s,a,s') = r$ whenever s' is closer (in whatever appropriate sense) to the goal than s, and $F(s,a,s') = 0$ otherwise, where r is some positive reward. Or, to encourage taking action a_1 in some set of states S_0, one might set $F(s,a,s') = r$ whenever $a = a_1, s \in S_0$, and $F(s,a,s') = 0$ otherwise.

One elementary but important property of this form of reward transformation is that it can generally be *implemented*: In many reinforcement learning applications, we are not explicitly given M as a tuple (S, A, T, γ, R), but are allowed to learn about M only through taking actions in the MDP and by observing the resulting state transitions and rewards. Given such access to M, we can simulate having the same type of access to M' simply by taking actions

[3]In the full paper, we will consider an even more general, not necessarily additive, form: $R'(s,a,s') = F(r,s,a,s')$ for arbitrary F, and where $r = R(s,a,s')$ is the reward we would have received in the original MDP M. Under the appropriate conditions, it turns out that, if we are to give optimality guarantees similar to those we will give here, then the only additional freedom this gives us in choosing shaping rewards is it allows us to rescale rewards by any fixed positive factor. Since this does not add any interesting richness to F, we defer this result to the full paper.

in M, and then "pretending" we observed reward $R(s,a,s') + F(s,a,s')$ whenever we actually observed reward $R(s,a,s')$ in M. Naturally, the simple reason that this works is that M and M' use the same actions, states and transition probabilities. Thus, online/offline model-based/model-free algorithms that may be applied to M may in general be readily applied to M' in the same way.

Since we are learning a policy for M' in the hope of using it in M, the question at hand is thus the following: For what forms of shaping-reward functions F can we guarantee that $\pi^*_{M'}$, the optimal policy in M', will also be optimal in M? The next section will answer this to a fair degree of generality.

3 Main results

In practical applications, we often do not exactly know T a priori (and may or may not know $R(s,a,s')$). Our goal is therefore, given S and A (and possibly R), to come up with a shaping-reward function $F : S \times A \times S \mapsto \mathbb{R}$ that is "good" and so that $\pi^*_{M'}$ will be optimal in M. In this section, we will give a form for F under which we can guarantee $\pi^*_{M'}$ will be optimal in M. We also provide a weak converse showing that, without further knowledge of T and R, this is the only type of shaping function that can always give this guarantee.

First focusing on the undiscounted case ($\gamma = 1$), let us try to gain some intuition about what F might give rise to the shaping "bug" pointed out in the Introduction. On Randløv and Altrøm's bicycle task, when the agent was rewarded for riding towards the goal but not punished for riding away from it, it learned to ride in a tiny circle and thereby obtain positive reward whenever it happened to be moving towards the goal. More generally, if there is some sequence of states $s_1, s_2, \ldots, s_n$ such that the agent can travel through them in a cycle ($s_1 \to s_2 \to \cdots \to s_n \to s_1 \to \cdots$), and gain net positive shaping-reward by doing so ($F(s_1,a_1,s_2) + \cdots + F(s_{n-1},a_{n-1},s_n) + F(s_n,a_n,s_1) > 0$), then it seems that the agent may be "distracted" from whatever it really should be trying to do (such as ride towards the goal,) and instead try to repeatedly go round this cycle.

To address this difficulty with cycles, a form for F that immediately comes to mind is to let F be a *difference of potentials*: $F(s,a,s') = \Phi(s') - \Phi(s)$, where Φ is some function over states. This way, $F(s_1,a_1,s_2) + \cdots + F(s_{n-1},a_{n-1},s_n) + F(s_n,a_n,s_1) = 0$, and we have eliminated the problem of cycles that "distract" the agent. Are there other ways to choose F? And aside from cycles, are there any other problems with shaping that we need to address? It turns out that, without more prior knowledge about T and R, such potential-based shaping functions F are the only F that will guarantee consistency with the optimal policy in M. Moreover, this turns out to be essentially all we need in order to make this guarantee. This is made formal in the following theorem:

Theorem 1 *Let any S, A, γ, and any shaping reward function $F : S \times A \times S \mapsto \mathbb{R}$ be given. We say F is a* **potential-based** *shaping function if there exists a real-valued function $\Phi : S \mapsto \mathbb{R}$ such that for all $s \in S - \{s_0\}, a \in A, s' \in S$,*

$$F(s,a,s') = \gamma\Phi(s') - \Phi(s), \tag{2}$$

(where $S - \{s_0\} = S$ if $\gamma < 1$). Then, that F is a potential-based shaping function is a necessary and sufficient condition for it to guarantee consistency with the optimal policy (when learning from $M' = (S,A,T,\gamma,R+F)$ rather than from $M = (S,A,T,\gamma,R)$), in the following sense:

- *(Sufficiency) If F is a potential-based shaping function, then every optimal policy in M' will also be an optimal policy in M (and vice versa).*
- *(Necessity) If F is not a potential-based shaping function (e.g. no such Φ exists satisfying Equation (2)), then there exist (proper) transition functions T and a reward function $R : S \times A \mapsto \mathbb{R}$, such that no optimal policy in M' is optimal in M.*

Also note the following: For the infinite-state case, if one were to choose some Φ to construct a potential-based shaping function, then for the formal results to go through, we really should demand that Φ be bounded, so that the shaping rewards F are also bounded (similar to the condition that R be bounded, in Section 2.1); this issue will be discussed again later. Note that for the finite-state case, this is a vacuous condition since Φ would, having a range of finite cardinality, automatically be bounded. Also, the necessity and sufficiency conditions above might seem a little more complicated than usual, and this is because there can be multiple optimal policies in M or in M'. Nevertheless, it should be clear that the quantifications used make this the strongest possible theorem of this form. The sufficiency condition says that so long as we use a potential-based F, then we are guaranteed any $\pi^*_{M'}$ we might be trying to learn will also be optimal in M. The necessity condition says that if we have no knowledge of T and R, then we must choose a potential-based F for learning in M', if we want to guarantee consistency with learning the optimal policy in M. (If we do have intimate knowledge of T, R, then the necessity condition does not say much, and it is possible that we might be able to use other shaping functions.)

The proof of necessity is given in Appendix A. Here, we only prove that Equation (2) is a sufficient condition: that if F is indeed of the form in (2), then we may guarantee that every optimal policy in M' will also be optimal in M. Again, we prove this result fully rigorously only for the case of finite $|S|$; the proof for infinite $|S|$ is nearly identical, but requires a little more care in justifying the use of the Bellman Equations.

Proof (of sufficiency): Let F be of the form given in (2). If $\gamma = 1$, then since replacing $\Phi(s)$ with $\Phi'(s) = \Phi(s) - k$ for any constant k would not change the shaping rewards F (which is a difference of these potentials), we may, by replacing $\Phi(s)$ with $\Phi(s) - \Phi(s_0)$ if necessary, assume without loss of generality that the Φ used to express F via (2) satisfies $\Phi(s_0) = 0$.

For the original MDP M, we know that its optimal Q-function Q^*_M satisfies the Bellman Equations (see e.g. [Sutton and Barto, 1998])

$$Q^*_M(s,a) = \mathrm{E}_{s' \sim P_{sa}(\cdot)}\left[R(s,a,s') + \gamma \max_{a' \in A} Q^*_M(s',a')\right]$$

Some simple algebraic manipulation then gives us

$$Q^*_M(s,a) - \Phi(s) = \mathrm{E}_{s'}\left[R(s,a,s') + \gamma\Phi(s') - \Phi(s) + \gamma \max_{a' \in A}\left(Q^*_M(s',a') - \Phi(s')\right)\right]$$

If we now define $\hat{Q}_{M'}(s,a) \triangleq Q^*_M(s,a) - \Phi(s)$ and substitute that and $F(s,a,s') = \gamma\Phi(s') - \Phi(s)$ back into the previous equation, we get

$$\begin{aligned}\hat{Q}_{M'}(s,a) &= \mathrm{E}_{s'}\left[R(s,a,s') + F(s,a,s') + \gamma \max_{a' \in A} \hat{Q}_{M'}(s',a')\right] \\ &= \mathrm{E}_{s'}\left[R'(s,a,s') + \gamma \max_{a' \in A} \hat{Q}_{M'}(s',a')\right]\end{aligned}$$

But this is exactly the Bellman equation for M'. For the undiscounted case, we moreover have $\hat{Q}_{M'}(s_0,a) = Q^*_M(s_0,a) - \Phi(s_0) = 0 - 0 = 0$. So, $\hat{Q}_{M'}(s,a)$ satisfies the Bellman equations for M', and must in fact be the unique optimal Q-function. Thus, $Q^*_{M'}(s,a) = \hat{Q}_{M'}(s,a) = Q^*_M(s,a) - \Phi(s)$, and the optimal policy for M' therefore satisfies

$$\begin{aligned}\pi^*_{M'}(s) &\in \arg\max_{a \in A} Q^*_{M'}(s,a) \\ &= \arg\max_{a \in A} Q^*_M(s,a) - \Phi(s) \\ &= \arg\max_{a \in A} Q^*_M(s,a)\end{aligned}$$

and is therefore also optimal in M. To show every optimal policy in M is also optimal in M', simply apply the same proof with the roles of M and M' interchanged (and using the shaping function $-F$). This completes the proof. $\square$

Corollary 2 *Under the conditions of Theorem 1, suppose that F does indeed take the form $F(s,a,s') = \gamma\Phi(s') - \Phi(s)$. Suppose further that $\Phi(s_0) = 0$ if $\gamma = 1$. Then for all $s \in S$, $a \in A$,*

$$Q^*_{M'}(s,a) = Q^*_M(s,a) - \Phi(s), \quad (3)$$

$$V^*_{M'}(s) = V^*_M(s) - \Phi(s). \quad (4)$$

Proof: (3) was proved in the sufficiency proof above; (4) follows immediately from the identity $V^*(s) = \max_{a \in A} Q^*(s,a)$. $\square$

Remark 1 (Robustness and learning): Although we have not proved it here, the identities in Corollary 2 actually hold for arbitrary policies π, not just the optimal policy: $V^\pi_{M'}(s) = V^\pi_M(s) - \Phi(s)$ (and similarly for Q-functions). A consequence of this is that potential-based shaping is *robust* in the sense that near-optimal policies are also preserved; that is, if we learn a near optimal policy π in M' (say, $|V^\pi_{M'}(s) - V^*_{M'}(s)| < \varepsilon$) using potential-based shaping, then π will also be near-optimal in M ($|V^\pi_M(s) - V^*_M(s)| < \varepsilon$). (To see this, apply the identity we just pointed out to policies π and to $\pi^*_M = \pi^*_{M'}$, and subtract.)

Remark 2 (All policies optimal under Φ): To better understand why potential-based F preserve optimal policies, it is worth noting if we have an MDP M that has a potential-based *reinforcement function* $R(s,a,s') = \gamma\Phi(s') - \Phi(s)$, then *any* policy is optimal in M. Thus, potential-based shaping functions are indifferent to policies, in the sense that they give us no reason to prefer any policy over any other; at an intuitive level, this accounts for why they do not give us any reason to prefer any policy other than π^*_M when we switch from M to M'.

The Theorem suggests that we choose shaping rewards of the form $F(s,a,s') = \gamma\Phi(s') - \Phi(s)$. In applications,

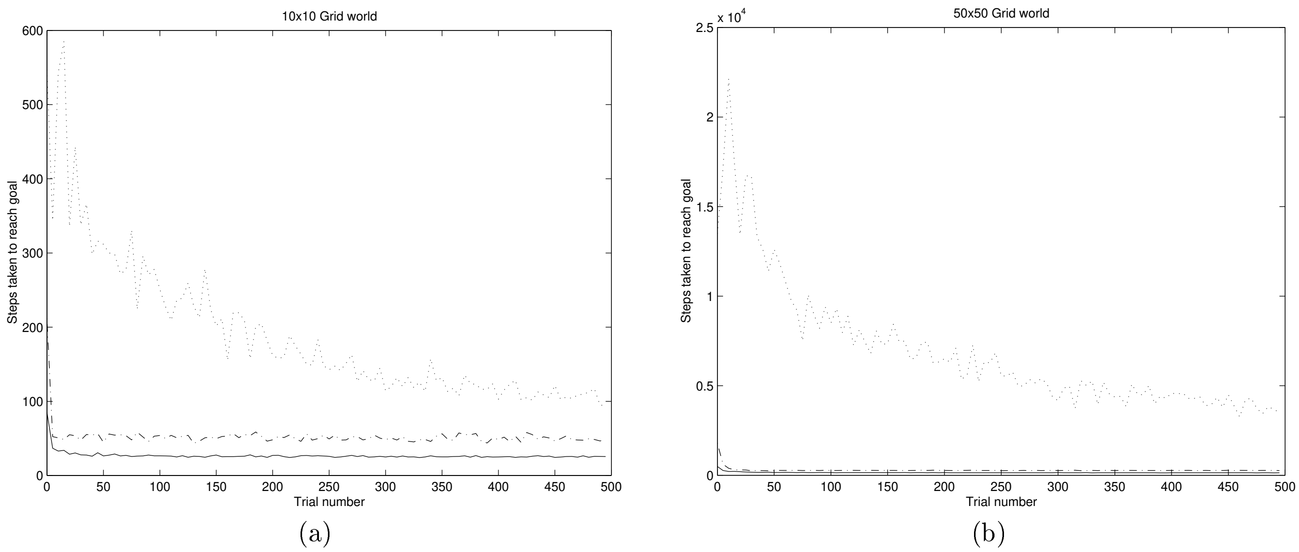


Figure 1: (a) Experiment with 10x10 grid-world. Plot of steps taken to goal vs. trial number. Dot is no shaping, dot-dash is $\Phi = 0.5\Phi_0$, solid is $\Phi = \Phi_0$. (b) Experiment with 50x50 grid-world.

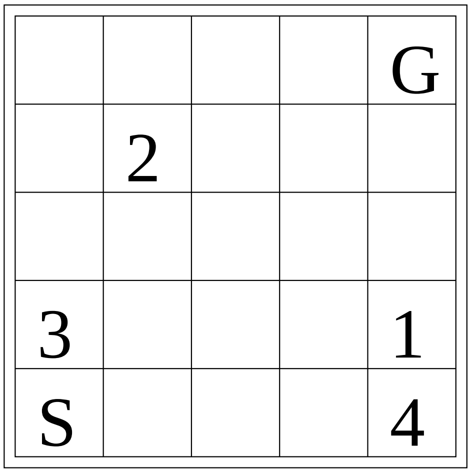
Φ should of course be chosen using expert knowledge about the domain. As to how one may do this, Corollary 2 suggests a particularly nice form for Φ, if we know enough about the domain to try choosing it as such. We see that if $\Phi(s) = V^*_M(s)$ (with $\Phi(s_0) = 0$ in the undiscounted case), then Equation (4) tells us that the value function in M' is $V^*_{M'}(s) \equiv 0$ — and this is a particularly easy value function to learn; even lacking a model of the world, all that would remain to be done would be to learn the non-zero Q-values. Though to avoid misconception, we also stress this is not the only way of choosing useful Φ, and that such shaping rewards can help significantly *even if* Φ *is far from* V^*_M (say in the sup-norm), such as by guiding exploration, etc., and we will see examples of this in the next section. But in any case, so long as we choose potential-based F, we have the guarantee that any (near-)optimal policy we learn in M' will also be (near-)optimal in M. Let us now turn our attention to some small experiments that demonstrate how potential-based shaping might be applied in practice.

4 Experiments

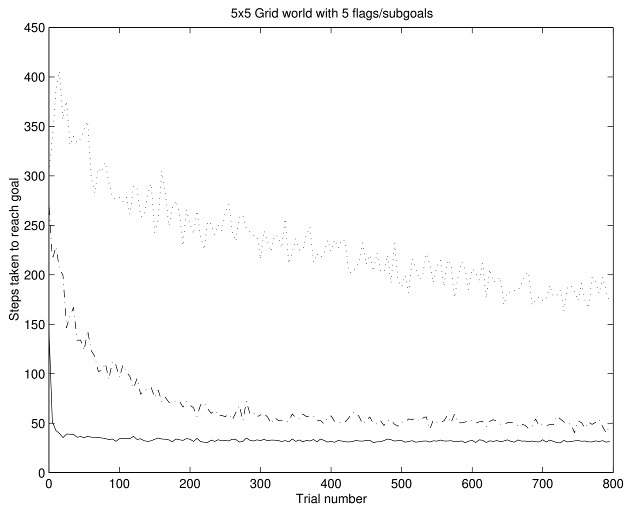
Much empirical work before us has convincingly justified the use of shaping [Mataric, 1994, Randløv and Alstrøm, 1998], and we will not bother to try to further justify its use. Here, our goal instead is to show how potential-style shaping functions fit into the picture, and to demonstrate how such shaping functions might be derived in practice.

Towards these goals, we chose for simplicity and clarity to use very simple grid-world domains to showcase the interesting aspects of potential-based shaping. The first domain was a shortest-path-to-goal 10x10 gridworld with start and goal states in opposite corners, no discounting, and a -1 per-step reinforcement. Actions are the 4 compass directions, and move 1 step in the intended direction 80% of the time and a random direction 20% of the time, and agent stays in the same place if it tries to walk off the grid. What might be a good shaping potential $\Phi(s)$? We had pointed out earlier that Equation (4) suggests $\Phi(s) = V^*_M(s)$ might be a good shaping potential. So let us now go through the type of reasoning that might suggest a crude estimate of V^*_M; by doing so, we hope to demonstrate how, with a little expert knowledge about distances and the location of the goal, similar reasoning may perhaps be used to similarly derive Φ for other minimum-cost-to-goal problems.

Upon trying to take a step towards the goal, we have an 80% chance of taking the desired step towards the goal, and a 20% chance of a random action. If we take a random action, then unless we are at the border of the gridworld, we are as likely to move towards as away from the goal. Hence, from most states, we would expect the optimal policy to make about 0.8 steps of (Manhattan distance) progress towards the goal per timestep, and a crude estimate of the expected number of steps needed to get to the goal from s would be MANHATTAN(s,GOAL)/0.8. Thus, we set



(a)



(b)

Figure 2: (a) 5x5 grid-world with 5 subgoals (including goal state), which must be visited in order $1, 2, 3, 4, G$. (b) Experiment with 5x5 grid-world with subgoals. Plot of steps taken to goal vs. trial number. Dot is no shaping, dot-dash is $\Phi = \Phi_0$, solid is $\Phi = \Phi_1$.

our estimate of the value function and therefore $\Phi(s)$ to be $\Phi_0(s) = \hat{V}_M(s) = -\text{MANHATTAN}(s,\text{GOAL})/0.8$. This is what we used as our guess of a "good" shaping function. Also, as a shaping-reward that would be quite far (in the sup-norm) from $V^*_{M'}(s)$, we also tried using $\Phi(s) = 0.5\Phi_0(s)$. The results of this first experiment[4] are shown in Figure 1a. (All experiments reported in this section are averages over 40 independent runs.) As can be readily seen, using either of these shaping functions significantly helped speed up learning. Moreover, it is worth re-stressing that even though $0.5\Phi_0$ is quite far from V^*_M, it still significantly helped the initial stages of learning. For a larger 50x50 grid-world, the results become even more dramatic: Figure 1b shows the result of the same experiment repeated on the larger grid. The plots for Φ_0 and $0.5\Phi_0$ are so low in the graph that they can barely be seen; learning without shaping is clearly losing hopelessly to the potential-based shaping algorithm.

Reiterating, the goal of these experiments was not to try to justify shaping — that has been done far more convincingly by others. Instead, what we have demonstrated here is a style of some very simple reasoning that, by putting together a distance-to-goal heuristic, has enabled us to pick a sensible Φ that dramatically sped up learning.

Next, another class of problems for which a similar style of reasoning might work is domains where we can assign subgoals. Consider the grid-world in Figure 2a, where we start in the lower-left hand corner, and must pick up a set of "flags" in sequence before going to the final goal state. Actions and rewards are the same as in the previous grid-world, and the state-space is expanded to keep track of the collected flags. Since each flag is a subgoal, it is tempting to choose F so that we are rewarded for visiting the subgoals. Let us now see how a potential-function style of reasoning can indeed lead us to choose such an F, and how Equation (4) further suggests magnitudes for the subgoal rewards.

With knowledge of the subgoal locations and using reasoning analogous to that suggested earlier (0.8 steps of progress per timestep, etc.), we may estimate the expected number of timesteps, say t, needed to reach the goal. If we imagine that each subgoal is about equally hard to reach from the previous one, then having reached the n-th subgoal, we would still have about $((5-n)/5)t$ steps to go. A slightly more refined argument changes this to $((5-n-0.5)/5)t$ steps (where 0.5 comes from the "typical case" where we are halfway between the n-th and $n+1$-st subgoals), and so our first choice of $\Phi(s)$ is $\Phi_0(s) = -((5-n_s-0.5)/5)t$, where n_s denotes the number of subgoals we have achieved when we are at s. Using this form of shaping-reward function, we see that $\Phi(s) = \Phi_0(s)$ jumps by

[4] Using Sarsa [Sutton and Barto, 1998], 0.10-greedy exploration, learning rate 0.02. Experiments with Sarsa(λ) also gave analogous results showing shaping significantly speeding up learning.

$t/5$ whenever we reach any subgoal (other than the final goal state), and so the shaping reward function $F(s,a,s') = \Phi(s') - \Phi(s)$ is giving $t/5$ reward for reaching each of these subgoals. This is exactly what our intuition had suggested might be a good shaping reward. For comparison, we also carried out this experiment using a more fine-tuned shaping reward that, similar to the previous grid-world experiments, explicitly estimated the remaining time-to-goal for each state and constructed the corresponding $\Phi_1(s) = \hat{V}_M(s)$ potential function. The result of these experiments are shown in Figure 2b, and we see that using our first crude shaping function Φ_0 has allowed us to significantly speed up learning over not using shaping (and the fine-tuned Φ_1 unsurprisingly gave even better performance). When repeating this experiment on larger domains or with more subgoals, the results (not reported here) become even more dramatic.

5 Discussion and Conclusions

We have shown necessary and sufficient conditions for a shaping function F to leave optimal policies invariant. Here are two easy generalizations worth mentioning: Aside from guaranteeing consistency while trying to learn the optimal policy, it is easy to show (by an argument similar to Remark 1 in Section 3) that potential-based F also work when trying to learn a good policy from within a *restricted* class of policies, such as in the framework studied in [Kearns et al., 1999] (and which for example includes the task of finding the best weights for a neural network mapping from states to actions). Also, for Semi-Markov decision processes (SMDPs) where actions take varying amounts of time to complete, Equation (2) unsurprisingly generalizes to $F(s,a,s',\tau) = e^{-\beta\tau}\Phi(s') - \Phi(s)$, where τ is the time the action took to complete, and β is the discount rate.

Finally, the "$\gamma\Phi(s') - \Phi(s)$" form also seems on the surface reminiscent of terms in some of the equations used in Advantage learning [Baird, 1994] and λ-policy iteration [Bertsekas and Tsitsiklis, 1996]. At a very crude level, it turns out that each of them may be thought of as trying to modify Φ so to gain some computational or representational advantage. If we consider the problem of modifying Φ, then trying to learn a rough shaping function seems to lead quite naturally to an algorithm for multi-scale value-function approximation; and although it may initially seem unusual to try to *learn* a shaping function, it is the multiscale "rough vs. fine" approximation aspect that this leads to which makes it possibly powerful;[5] this will be the subject of future work.

In this paper, we have shown that potential-based shaping rewards $\gamma\Phi(s') - \Phi(s)$ leave (near-)optimal policies unchanged. Moreover, this was proved to be the only type of shaping that can guarantee such invariance unless we make further assumptions about the MDP. But just as some practitioners use discounting even on undiscounted problems (perhaps to improve convergence of algorithms), we believe that future experience with potential-style shaping rewards may also lead one to occasionally try shaping rewards that are inspired by potentials, but which are perhaps not strictly of the form we have given. For example, in analogy to using discounting even on undiscounted problems, it is conceivable that for certain problems, it may be easier for an expert to propose a potential Φ for an "undiscounted" shaping function $\Phi(s') - \Phi(s)$, even when $\gamma \neq 1$. Even though our theorem may no longer guarantee optimality in this case, such a shaping function may, purely from an engineering point of view, still be worth trying — judiciously and with care. In the same spirit, whereas our regularity conditions had demanded using bounded Φ, it is also plausible that some practitioners might want to try certain unbounded Φ. Naturally, if expert knowledge about the domain is available, then non-potential shaping functions might also be fully appropriate.

As guidelines for choosing shaping functions, we have suggested a distance-based heuristic and a subgoal-based heuristic for choosing potentials; because shaping is often crucial to making learning tractable, we believe the task of finding good shaping functions will be a problem of increasing importance.

Acknowledgments

A. Ng is supported by a Berkeley Fellowship. This work was also supported in part by ARO MURI grant DAAH04-96-1-0341, ONR grant N00014-97-1-0941, and NSF grant ECS-9873474.

References

[Baird, 1994] Baird, L. C. (1994). Reinforcement Learning in continuous time: Advantage updating.

[5] This also relates to the observation that something like a learned shaping reward seems to be operating psychologically—e.g., the capture of a piece in chess operates as a reward even though the underlying MDP has rewards only for checkmate.

In *Proceedings of the International Conference on Neural Networks.*

[Bertsekas, 1995] Bertsekas, D. P. (1995). *Dynamic Programming and Optimal Control, Volume II.* Athena Scientific.

[Bertsekas and Shreve, 1978] Bertsekas, D. P. and Shreve, S. E. (1978). *Stochastic Optimal Control: The Discrete Time Case.* Academic Press.

[Bertsekas and Tsitsiklis, 1996] Bertsekas, D. P. and Tsitsiklis, J. N. (1996). *Neuro-dynamic Programming.* Athena Scientific.

[Dorigo and Colombetti, 1994] Dorigo, M. and Colombetti, M. (1994). Robot shaping: Developing autonomous agents through learning. *Artificial Intelligence*, 71(2):321–370.

[Hernández-Lerma, 1989] Hernández-Lerma, O. (1989). *Adaptive Markov Control Processes.* Springer-Verlag.

[Kearns et al., 1999] Kearns, M., Mansour, Y., and Ng, A. Y. (1999). Approximate planning in large POMDPs via reusable trajectories. *(Preprint).*

[Mataric, 1994] Mataric, M. J. (1994). Reward functions for accelerated learning. In *Proceedings of the Eleventh International Conference on Machine Learning.* Morgan Kaufmann.

[Randløv and Alstrøm, 1998] Randløv, J. and Alstrøm, P. (1998). Learning to drive a bicycle using reinforcement learning and shaping. In *Proceedings of the Fifteenth International Conference on Machine Learning.* Morgan Kaufmann.

[Russell, 1998] Russell, S. (1998). Learning agents for uncertain environments (extended abstract). In *Proceedings of the Eleventh Annual ACM Workshop on Computational Learning Theory (COLT-98)*, Madison, Wisconsin. ACM Press.

[Rust, 1994] Rust, J. (1994). Do people behave according to Bellman's principal of optimality? Submitted to Journal of Economic Perspectives.

[Saksida et al., 1997] Saksida, L., Raymond, S., and Touretzky, D. (1997). Shaping robot behaviour using principles from instrumental conditioning. *Robotics and Autonomous Systems*, 22(3–4):231–249.

[Singh and Yee, 1994] Singh, S. and Yee, R. (1994). An upper bound on the loss from approximate optimal-value functions. *Machine Learning*, 16:227–233.

[Sutton and Barto, 1998] Sutton, R. S. and Barto, A. G. (1998). *Reinforcement Learning: An Introduction.* MIT Press.

[von Neumann and Morgenstern, 1944] von Neumann, J. and Morgenstern, O. (1944). *Theory of Games and Economic Behavior.* Princeton University Press, Princeton, New Jersey, first edition.

[Williams and Baird, 1994] Williams, R. J. and Baird, L. C. (1994). Tight performance bounds on greedy policies based on imperfect value functions. In *Proceedings of the Tenth Yale Workshop on Adaptive and Learning Systems.*

Appendix A: Proof of necessity

In this Appendix, we sketch the proof of the necessity part of Theorem 1. For brevity, we give the proof only for the case of $|A| = 2$; the generalization is obvious but more tedious. We begin with the following Lemma.

Lemma 3 *If there exists $s \in S - \{s_0\}, s' \in S$ and $a, a' \in A$ such that $F(s,a,s') \neq F(s,a',s')$, then there exists (proper) transition functions T and a reward function R such that no optimal policy in M' is optimal in M.*

Proof (Sketch, Lemma 3): Assume without loss of generality that $F(s,a,s') > F(s,a',s')$, and let $\Delta = F(s,a,s') - F(s,a',s') > 0$. In the undiscounted case, also assume for simplicity that $s \neq s'$. (When $\gamma = 1$, the proof for $s = s'$ is nearly the same, but having to ensure properness just makes it much more tedious.) We then construct M as follows: Let $P_{sa}(s') = P_{sa'}(s') = 1.0$, and let $R(s,a,s') = 0$ and $R(s,a',s') = \Delta/2$. Clearly $\pi^*_M(s) = a'$. On the other hand, since $R' = R + F$, we have $R'(s,a,s') = F(s,a,s')$ and $R'(s,a',s') = \Delta/2 + F(s,a',s') = F(s,a,s') - \Delta/2 < R'(s,a,s')$, and hence $\pi^*_{M'}(s) = a$. □

We are now ready to show the main necessity result.

Proof (of necessity). Assume F is not potential-based. We need to show we can construct T, R such that no optimal policy $\pi^*_{M'}$ in M' is also optimal in M. By Lemma 3, if $F(s,a,s')$ depends on a, we are done; hence we need only consider shaping functions

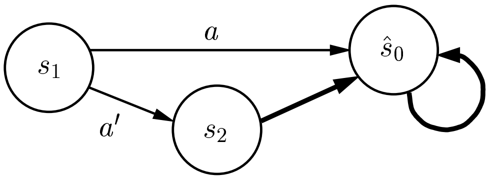


Figure 3: The unlabeled thick edges correspond to both actions. All edges have probability 1. The edge $(s_1, a, \hat{s}_0)$ carries a reward $\Delta/2$, and all other edges have zero reward.

of the form $F(s, a, s') = F(s, s')$ (which do not depend on a).

If $\gamma = 1$, let $\hat{s}_0 = s_0$ be the distinguished absorbing state; otherwise let $\hat{s}_0$ be some fixed state. Noting that constant offsets of the reward do not affect the optimal policy when $\gamma < 1$, we may, by replacing all $F(s, s')$ with $F(s, s') - F(\hat{s}_0, \hat{s}_0)$ if necessary, assume without loss of generality that $F(\hat{s}_0, \hat{s}_0) = 0$. Now define $\Phi(s) = -F(s, \hat{s}_0)$ for all s. By assumption of F not being potential-based, there exists s_1, s_2 such that $\gamma\Phi(s_2) - \Phi(s_1) \neq F(s_1, s_2)$ (let us assume $s_1, s_2, \hat{s}_0$ are distinct; the other cases are either impossible or handled similarly). We then construct M in the following way (still assuming $|A| = 2$). From state s_1, let $P_{s_1 a}(\hat{s}_0) = P_{s_1 a'}(s_2) = 1.0$, and from states s_2 and $\hat{s}_0$ let both actions a and a' lead to $\hat{s}_0$ with probability 1. Also define $\Delta = F(s_1, s_2) + \gamma F(s_2, \hat{s}_0) - F(s_1, \hat{s}_0)$ and let $R(s_1, a, \hat{s}_0) = \Delta/2$, $R(\cdot, \cdot, \cdot) = 0$ elsewhere. This model is illustrated in Figure 3. Then we have

$$\begin{aligned}
Q^*_M(s_1, a) &= \frac{\Delta}{2} \\
Q^*_M(s_1, a') &= 0 \\
Q^*_{M'}(s_1, a) &= \frac{\Delta}{2} + F(s_1, \hat{s}_0) \\
&= F(s_1, s_2) + \gamma F(s_2, \hat{s}_0) - \frac{\Delta}{2} \\
Q^*_{M'}(s_1, a') &= F(s_1, s_2) + \gamma F(s_2, \hat{s}_0),
\end{aligned}$$

where we have relied on the fact that $V^*_M(\hat{s}_0) = V^*_{M'}(\hat{s}_0) = 0$ by construction. Hence

$$\begin{aligned}
\pi^*_M(s_1) &= \begin{cases} a & \text{if } \Delta > 0, \\ a' & \text{otherwise} \end{cases} \\
\pi^*_{M'}(s_1) &= \begin{cases} a' & \text{if } \Delta > 0, \\ a & \text{otherwise} \end{cases}
\end{aligned}$$

□